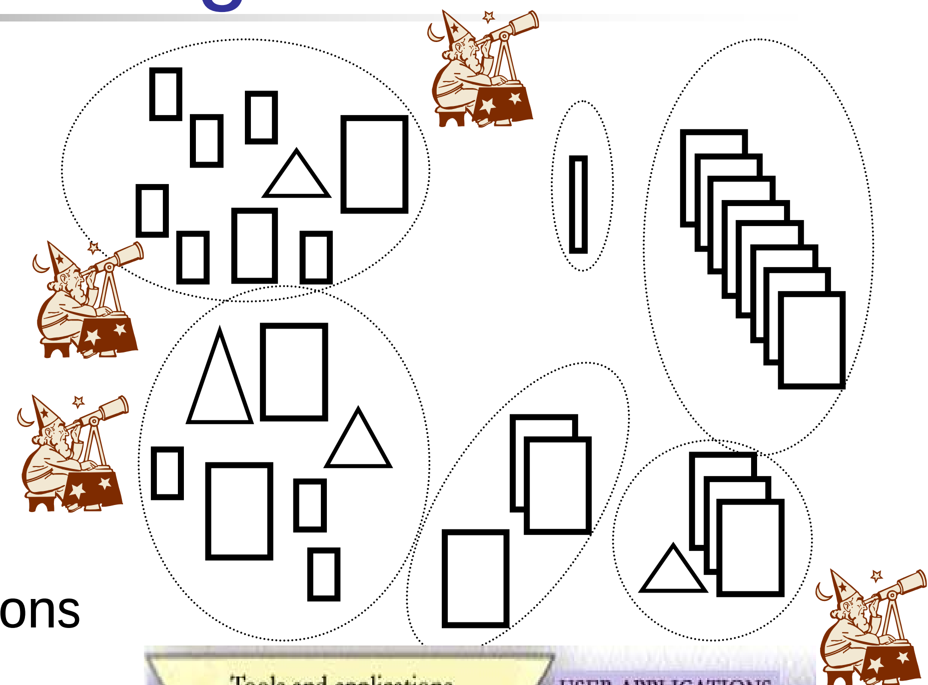
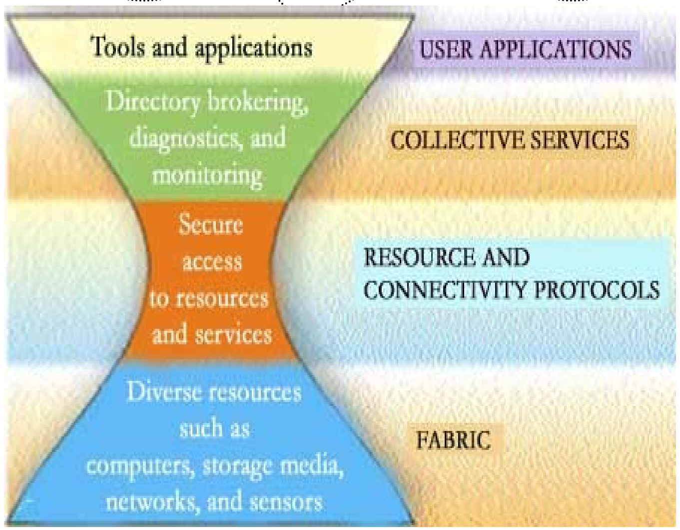
Filecules and Small Worlds in the DZero Workload: Characteristics and Significance

Adriana Iamnitchi
anda@cse.usf.edu
Computer Science & Engineering
University of South Florida

Grid: Resource-Sharing Environment

- Users:
  - 1000s from 10s institutions
  - Well-established communities
- Resources:
  - Computers, data, instruments, storage, applications
  - Owned/administered by institutions
- Applications: data- <u>and</u> compute-intensive processing
- Approach: common infrastructure





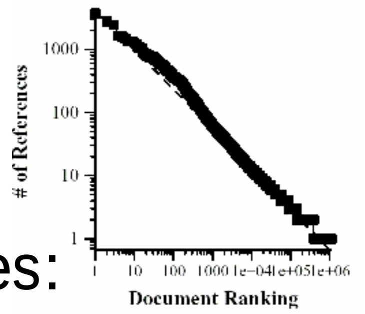
The Problem

- We have now:
  - Mature grid deployments running in production mode
- We do not have yet:

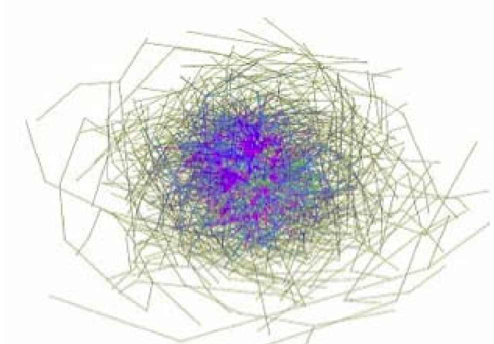
  - Quantitative characterization of real workloads.
    - How many files, how much input data per process, etc.
  - And thus, benchmarks, workload models, reproducible results
- Costs:
  - Local solutions, often replicating work
  - “Temporary” solutions that become permanent
  - Far from optimal solutions
  - Impossible to compare alternatives on relevant workloads

Still, Why Should We Care?

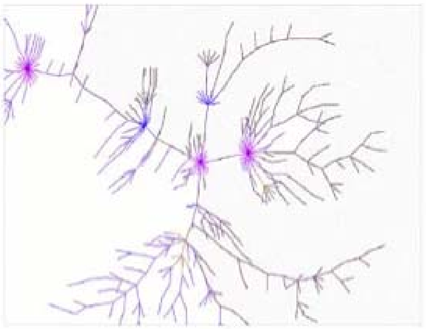
- Impossibility results, high costs: Tradeoffs are necessary
  - Solution: Select tradeoffs based on
    - User requirements (of course)
    - Usage patterns



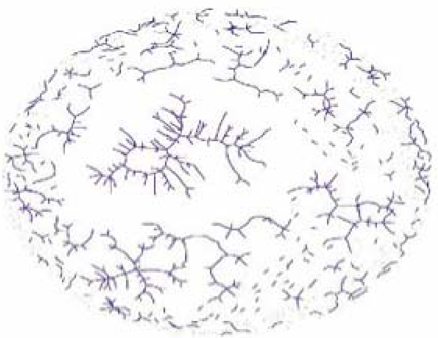
- Patterns exist and can be exploited. Examples:
  - Zipf distribution for request popularity (web caching) Breslau et al., Infocom'99
  - Network topology:



Partial Topology



Random 30% die



Targeted 4% die

from Saroiu *et al.*, *MMCN* 2002

This Presentation

- ...characterizes workloads from DZero from the perspective of data management
  - Data is the main resource shared in many grids
  - High-energy physics domain
  - Potentially representative for other domains
- ...proposes a data abstraction *(filecule)* relevant to multi-file data processing
- ...identifies a novel pattern *(small-world file sharing)* relevant to data sharing
- ...shows benefits via experiments
- and invites your comments and suggestions.

The DØ Experiment

- High-energy physics data grid
- 72 institutions, 18 countries, 500+ physicists
- Detector Data
  - 1,000,000 Channels
  - Event rate ~50 Hz
  - So far, 1.9 PB of data (Update?)
- Data Processing
  - Signals: physics events
  - Events about 250 KB, stored in files of ~1GB
  - Every bit of raw data is accessed for processing/filtering
  - Past year overall: 0.6 PB (Update?)
- DØ:
  - ... processes PBs/year
  - ... processes 10s TB/day
  - ... uses 25% – 50% remote computing

DØ Workload Characterization

Joint work with

Shyamala Doraimani (USF) and
Gabriele Garzoglio (FNAL)

DØ Traces (thanks to Ruth and Gabriele)

- Traces from January 2003 to May 2005
- 234,000 jobs, 561 users, 34 domains, 1.13 million files accessed
- 108 input files per job on average
- Detailed data access information about half of these jobs (113,062)

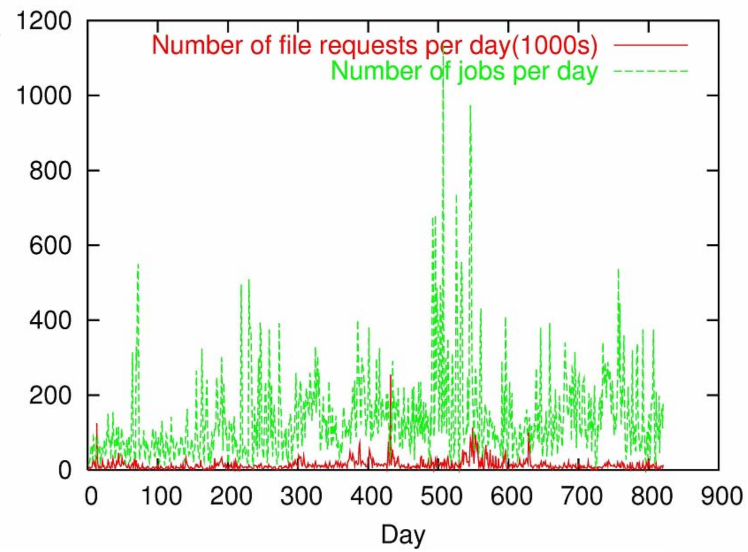


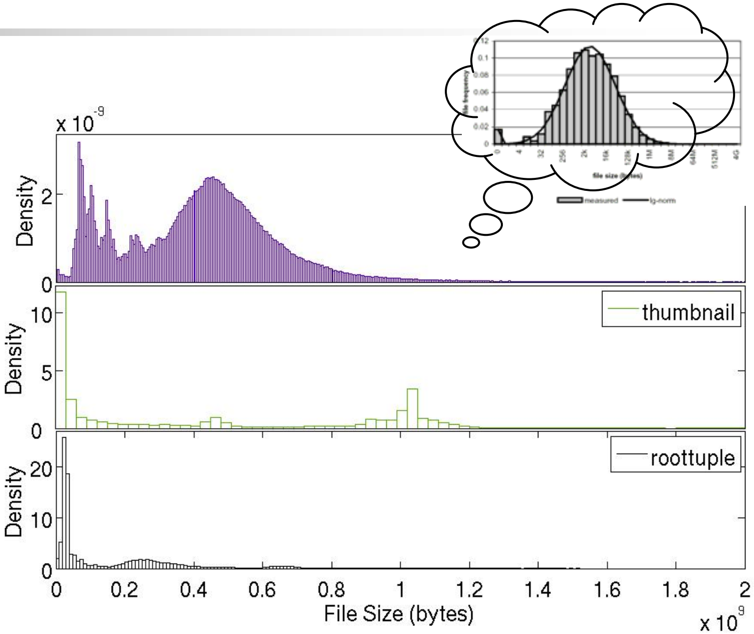
Table 1. Characteristics of traces analyzed per data tier.

Data Tier	Users	Jobs	Files	Input/Job (MB)	Time/Job (hours)
Reconstructed	320	17898	515677	36371	11.01
Root-tuple	63	1307	60719	83041	13.68
Thumbnail	449	94625	428610	53619	4.89
Others	435	120962	N/A	N/A	7.68
All	561	233792	N/A	N/A	6.87

Contradicts Traditional Models

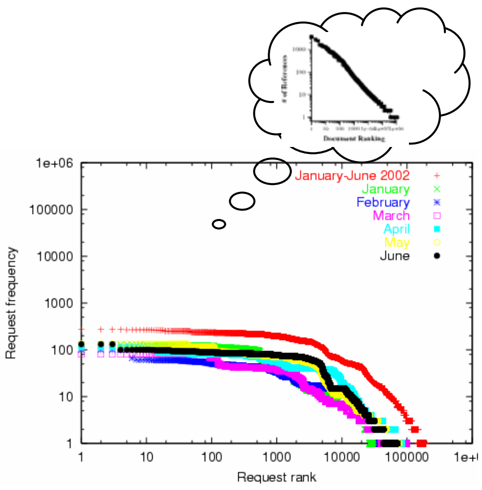
File size distribution

- Expected: log-normal. Why not?
  - Deployment decisions
  - Domain specific
  - Data transformation



File popularity distribution

- Expected: Zipf. Why not? (speculations):
- Scientific data is uniformly interesting
- User community is relatively small

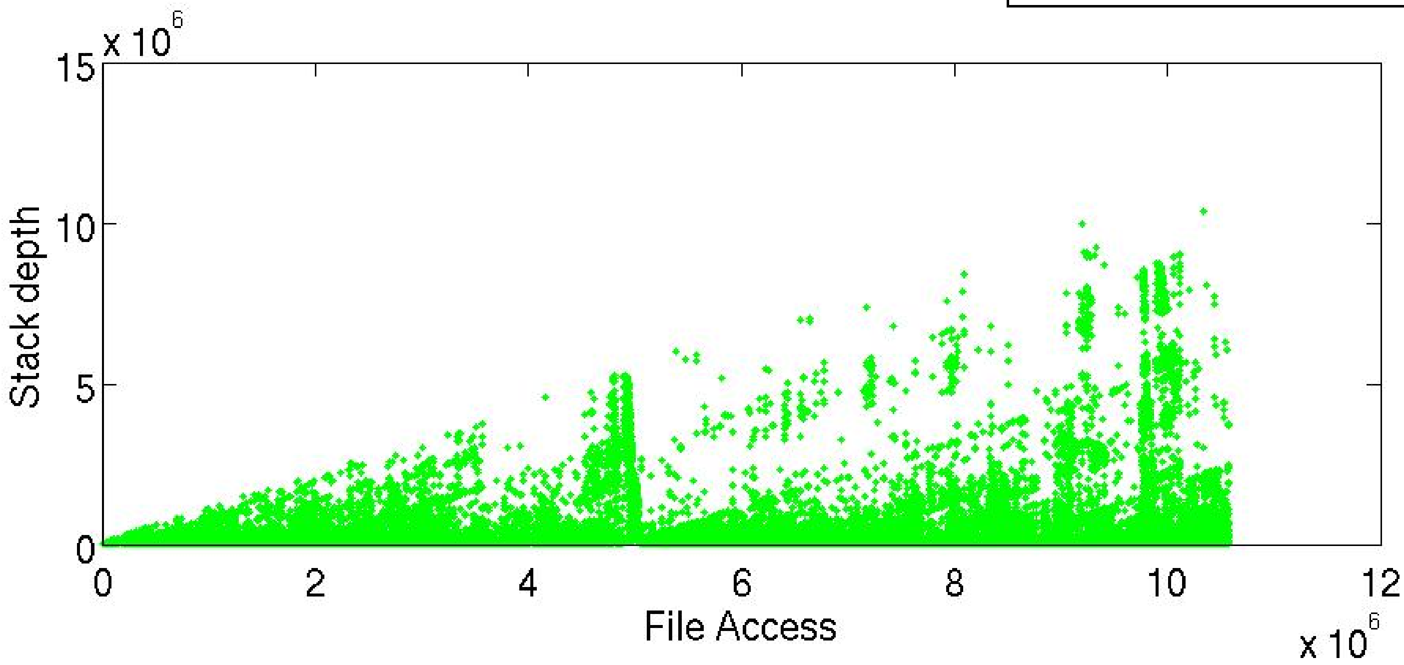


Time Locality

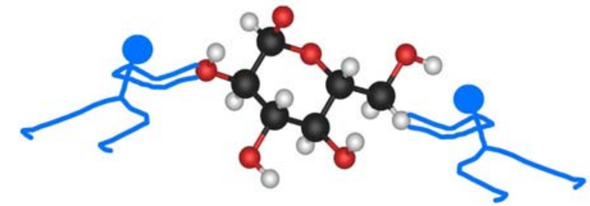
Stack-depth analysis

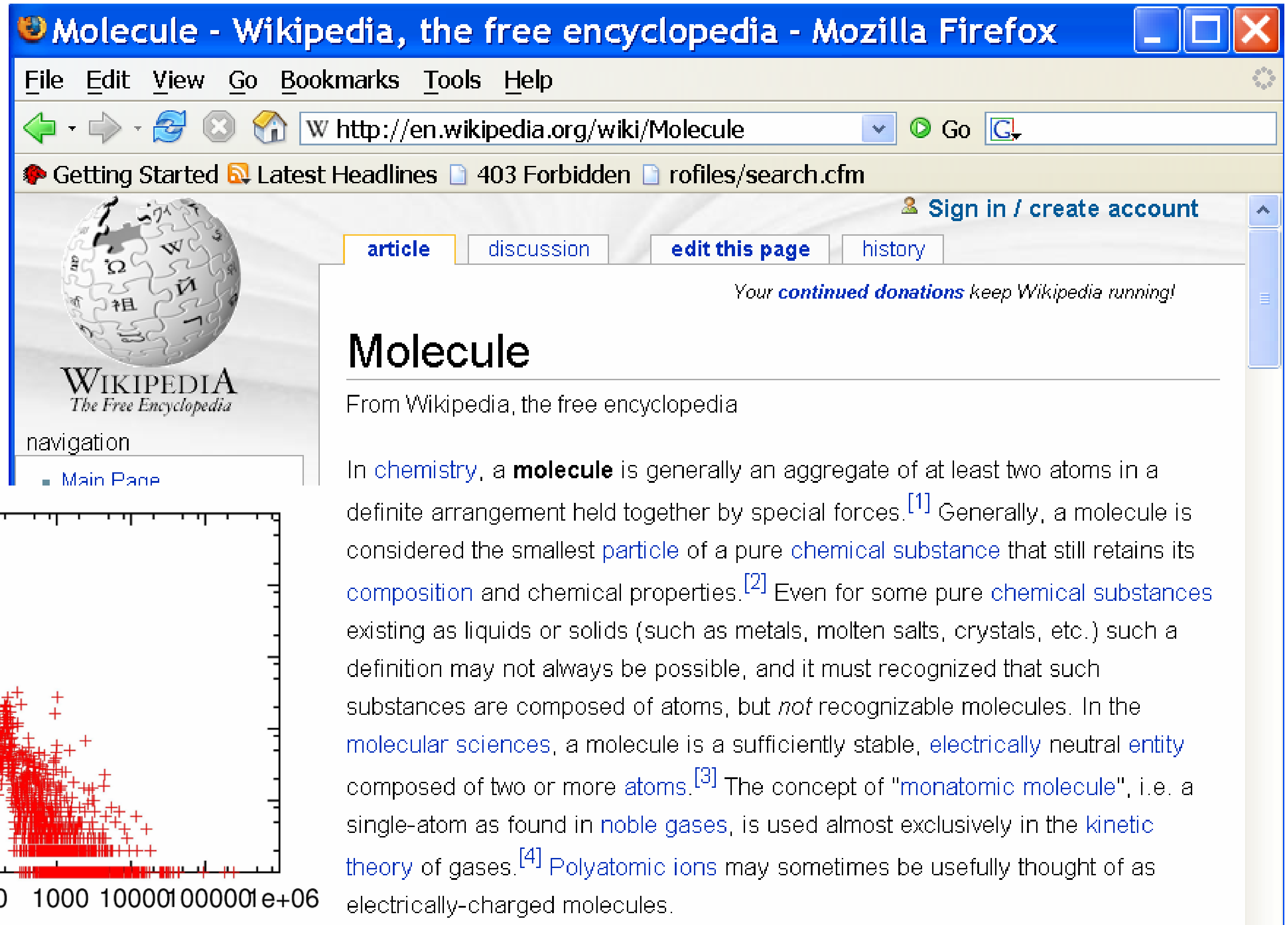
- Good temporal locality
- (to be used in cache replacement algorithms)

Measure	Value
Maximum	946,600
1 percentile	85
10 percentile	960
50 percentile (Median)	12,260
90 percentile	90,444
Standard Deviation	79,300



Filecules: Intuition





Molecule - Wikipedia, the free encyclopedia - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://en.wikipedia.org/wiki/Molecule Go

Getting Started Latest Headlines 403 Forbidden rofiles/search.cfm

Sign in / create account

article discussion edit this page history

WIKIPEDIA
The Free Encyclopedia

navigation

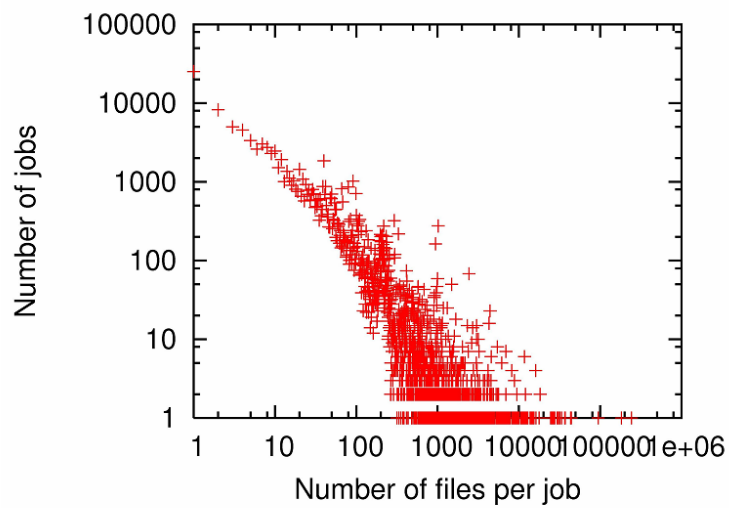
- Main Page

*Your **continued donations** keep Wikipedia running!*

Molecule

From Wikipedia, the free encyclopedia

In chemistry, a **molecule** is generally an aggregate of at least two atoms in a definite arrangement held together by special forces.[1] Generally, a molecule is considered the smallest particle of a pure chemical substance that still retains its composition and chemical properties.[2] Even for some pure chemical substances existing as liquids or solids (such as metals, molten salts, crystals, etc.) such a definition may not always be possible, and it must recognized that such substances are composed of atoms, but *not* recognizable molecules. In the molecular sciences, a molecule is a sufficiently stable, electrically neutral entity composed of two or more atoms.[3] The concept of "monatomic molecule", i.e. a single-atom as found in noble gases, is used almost exclusively in the kinetic theory of gases.[4] Polyatomic ions may sometimes be usefully thought of as electrically-charged molecules.



Filecules: General Characteristics

Table 2. Characteristics of analyzed traces per location.

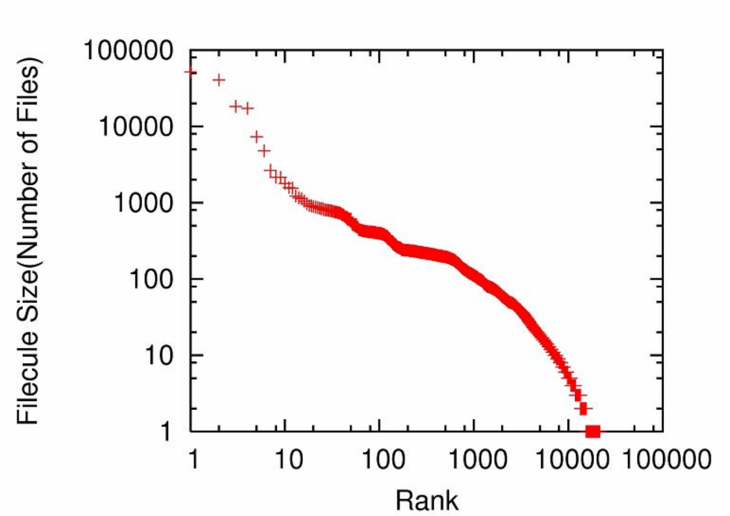
Domain	Jobs	Submission nodes	Sites	# users	# filecules	# files	Total data (GB)
.gov	3319711	12	1	466	95234	945031	4930850
.de	390186	5	4	23	33403	100257	268815
.uk	131760	8	4	21	23876	62427	117097
.edu	54672	18	12	32	14504	36868	41081
.cz	7400	1	1	1	4789	7660	9869
.ca	5719	5	2	4	649	8937	22341
.fr	5086	2	1	11	1767	18215	23958
.nl	3854	3	2	8	888	38812	44012
.mx	146	1	1	1	32	1589	349
.br	12	2	2	2	2	2	2
.cn	4	1	1	2	2	62	31
.in	3	1	1	2	2	2	0.70

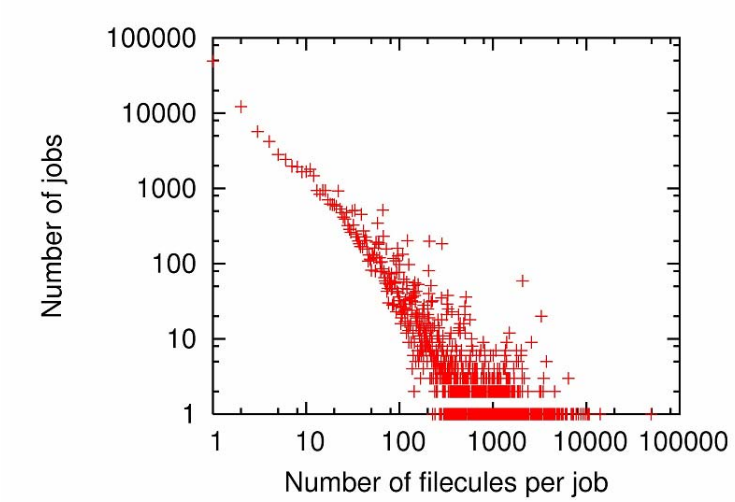
Filecules in High-Energy Physics: Characteristics and Impact on Resource Management, Adriana Iamnitchi, Shyamala Doraimani, Gabriele Garzoglio, HPDC'06

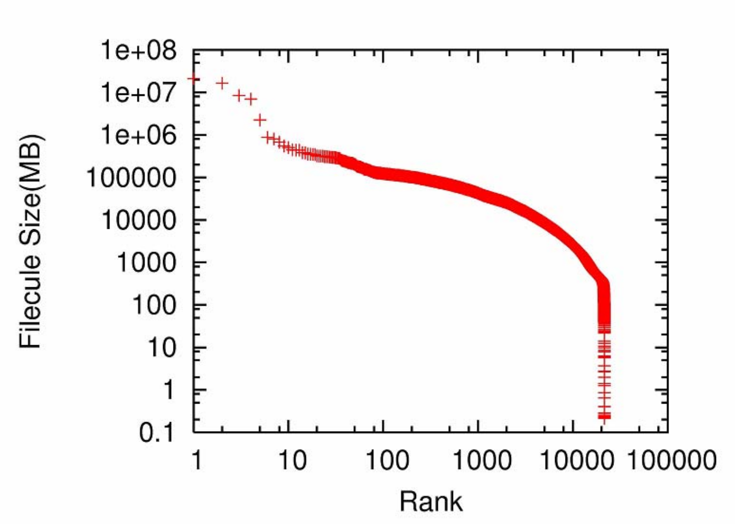
Filecules: Size

Filecules of different sizes:

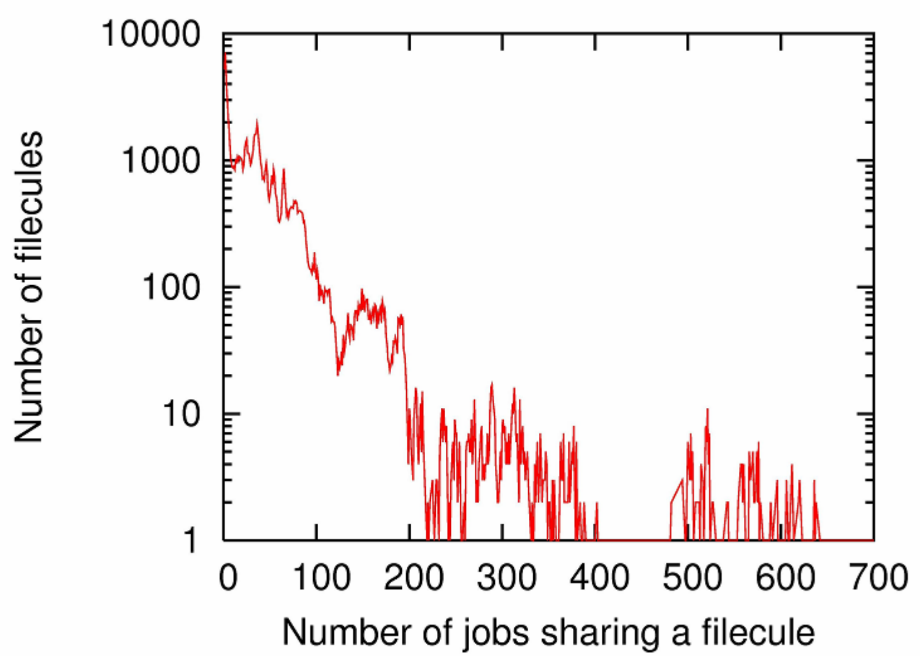
- Largest filecule:17 TB or 51,841 files
- 28% mono-file filecules

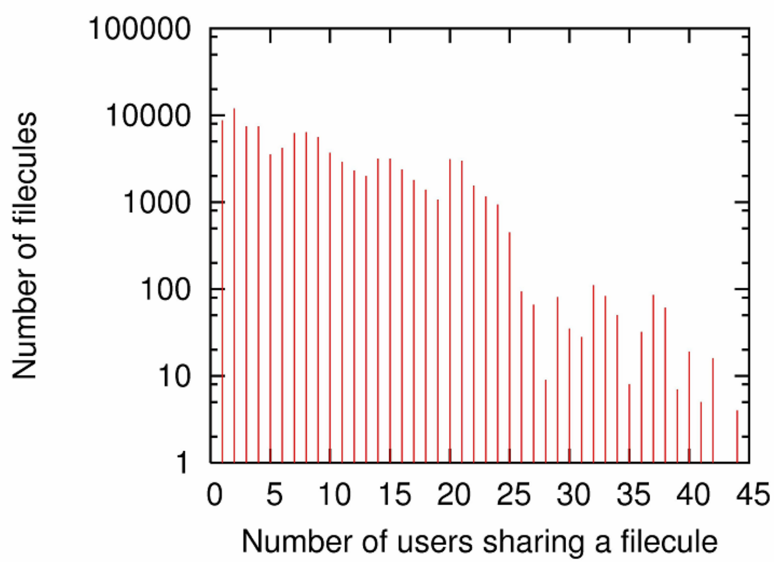


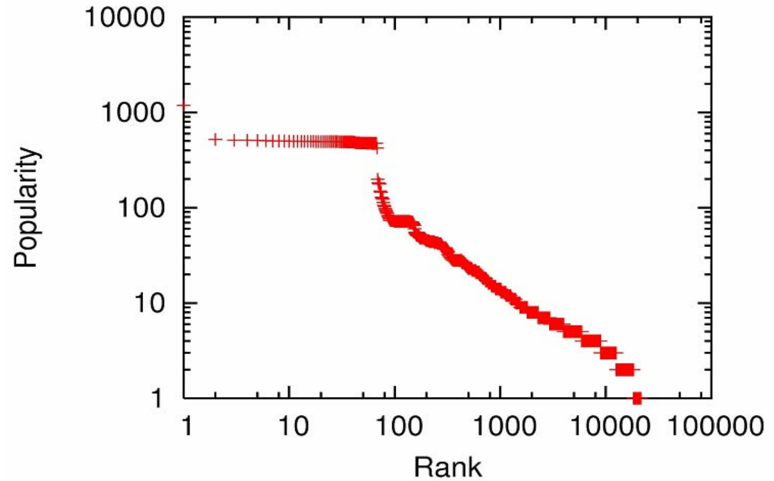




Filecules: Popularity





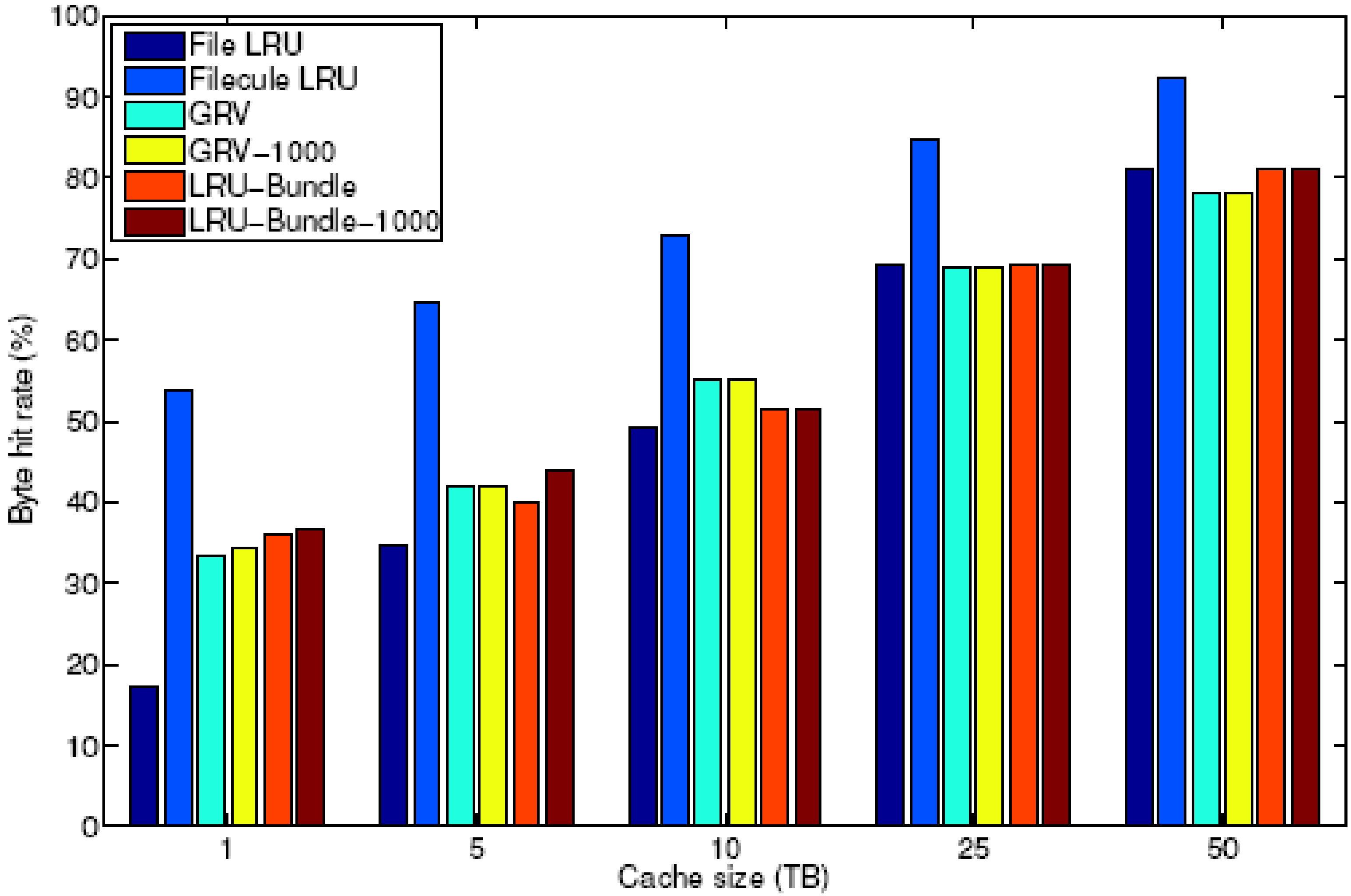


Consequences for Caching

- Use filecule membership for prefetching
  - When a file is missing from the local cache, prefetch the entire filecule
- Use time locality in cache replacement
  - Least Recently Used (classic algorithm)
- Implemented:
  - LRU with files and LRU with filecules
  - Greedy Request Value: prefetching + job reordering
    - Does not exploit temporal locality
    - Prefetching based on cache content
  - Our variant of LRU with filecules and job reordering

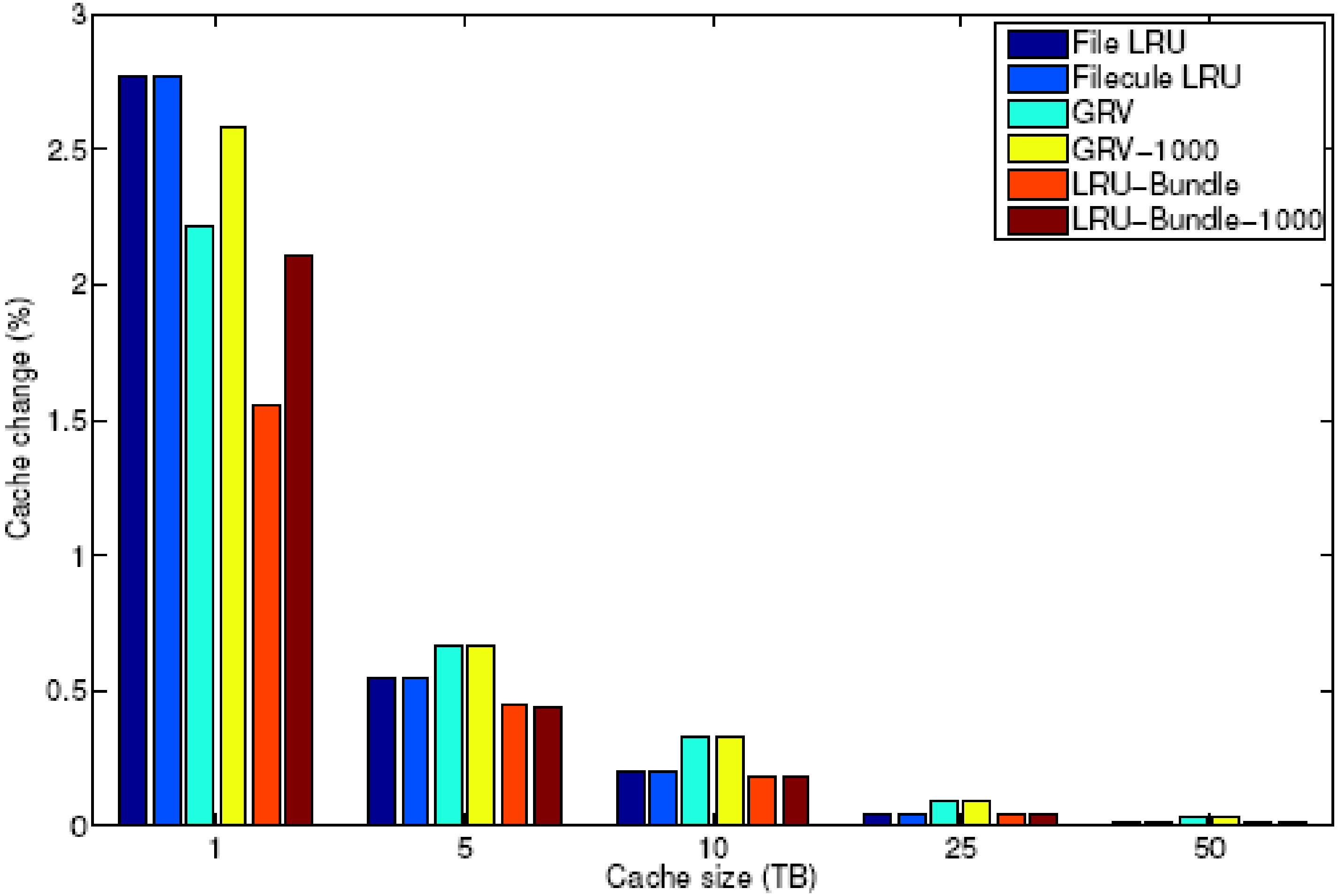
E. Otoo, et al. Optimal file-bundle caching algorithms for data-grids. In *SC '04*

Comparison: Caching Algorithms (1)



Comparison: Caching Algorithms (2)

% of cache change is a measure of transfer costs.



Summary Part 1

- Revisited traditional workload models
  - Generalized from file systems, the web, etc.
  - Some confirmed (temporal locality), some infirmed (file size distribution and popularity)
- Compared caching algorithms on D0 data:
  - Temporal locality is relevant
  - Filecules guide prefetching

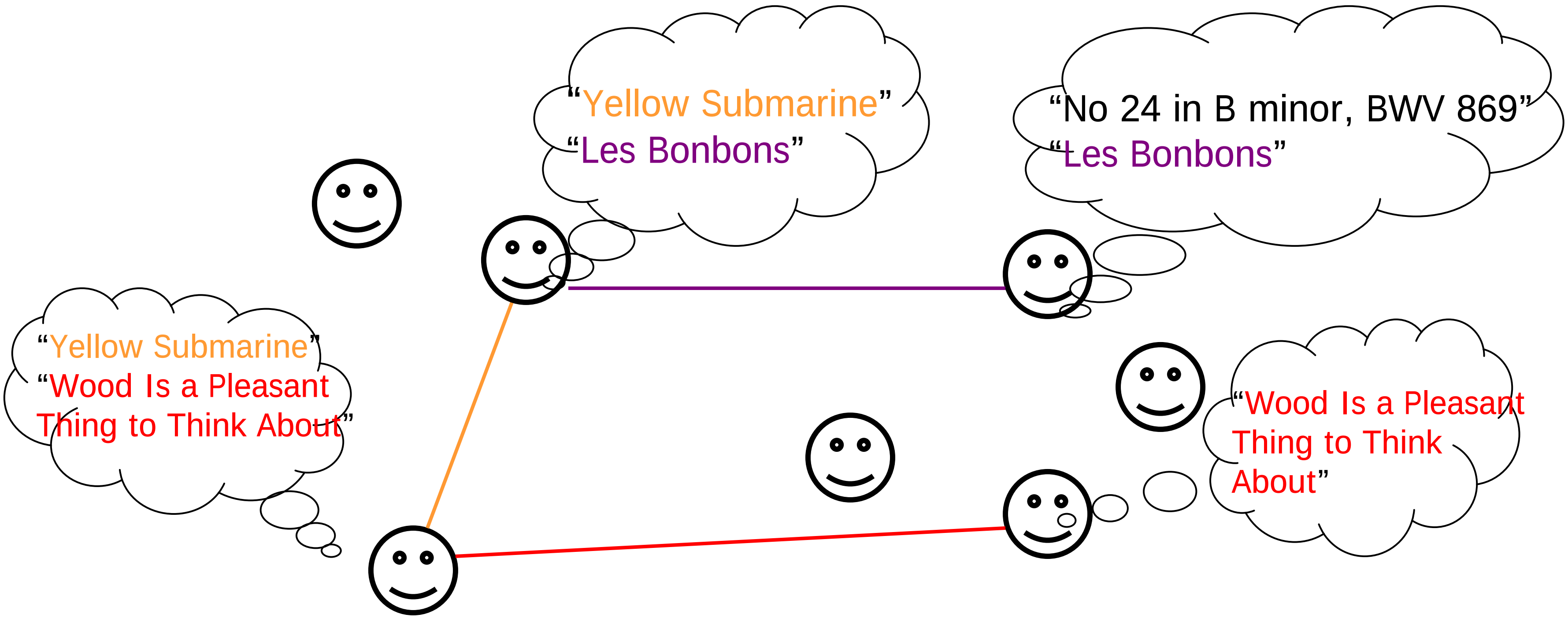
Metric	Algorithm with the best performance
Byte hit rate	Filecule LRU
Percentage of cache change	LRU-Bundle
Job Waiting Time	GRV
Queue Length	GRV
Scheduling Overhead	File LRU and Filecule LRU

Filecules and Small Worlds in Scientific Communities: Characteristics and Significance

Joint work with

Matei Ripeanu (UBC) and

Ian Foster (ANL and UChicago)

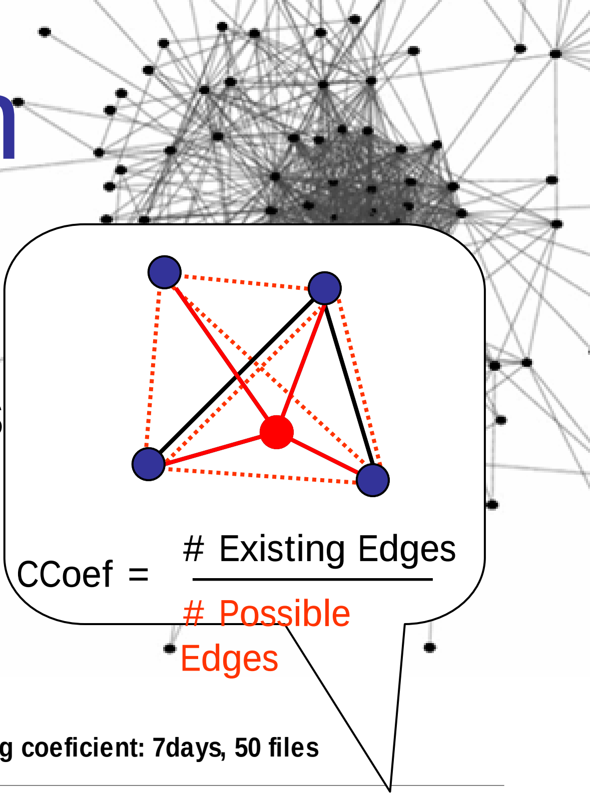


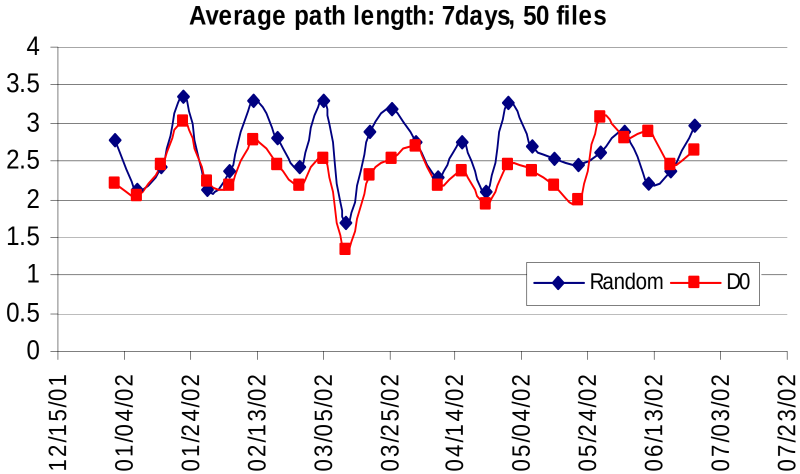
New metric: The Data-Sharing Graph $G_m^T(V, E)$:

- V is set of users active during interval T
- An edge in E connects users that asked for at least m common files within T

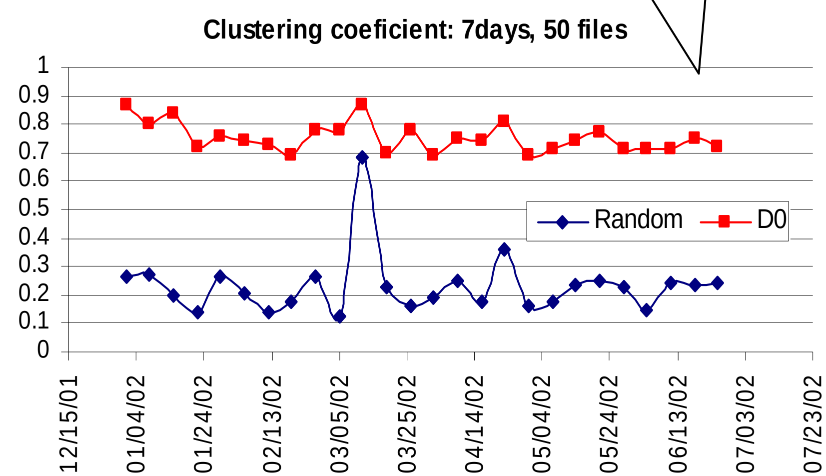
The DØ Collaboration

6 months of traces (January – June 2002)
300+ users, 2 million requests for 200K files





Small average path length

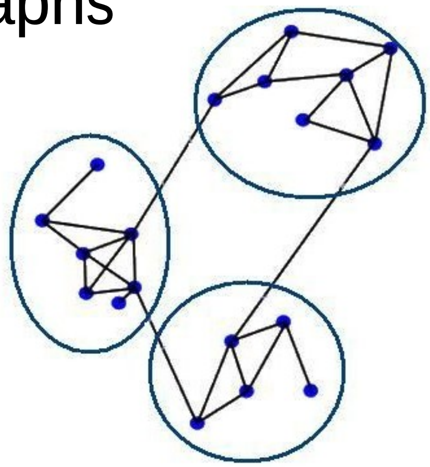


Small World!

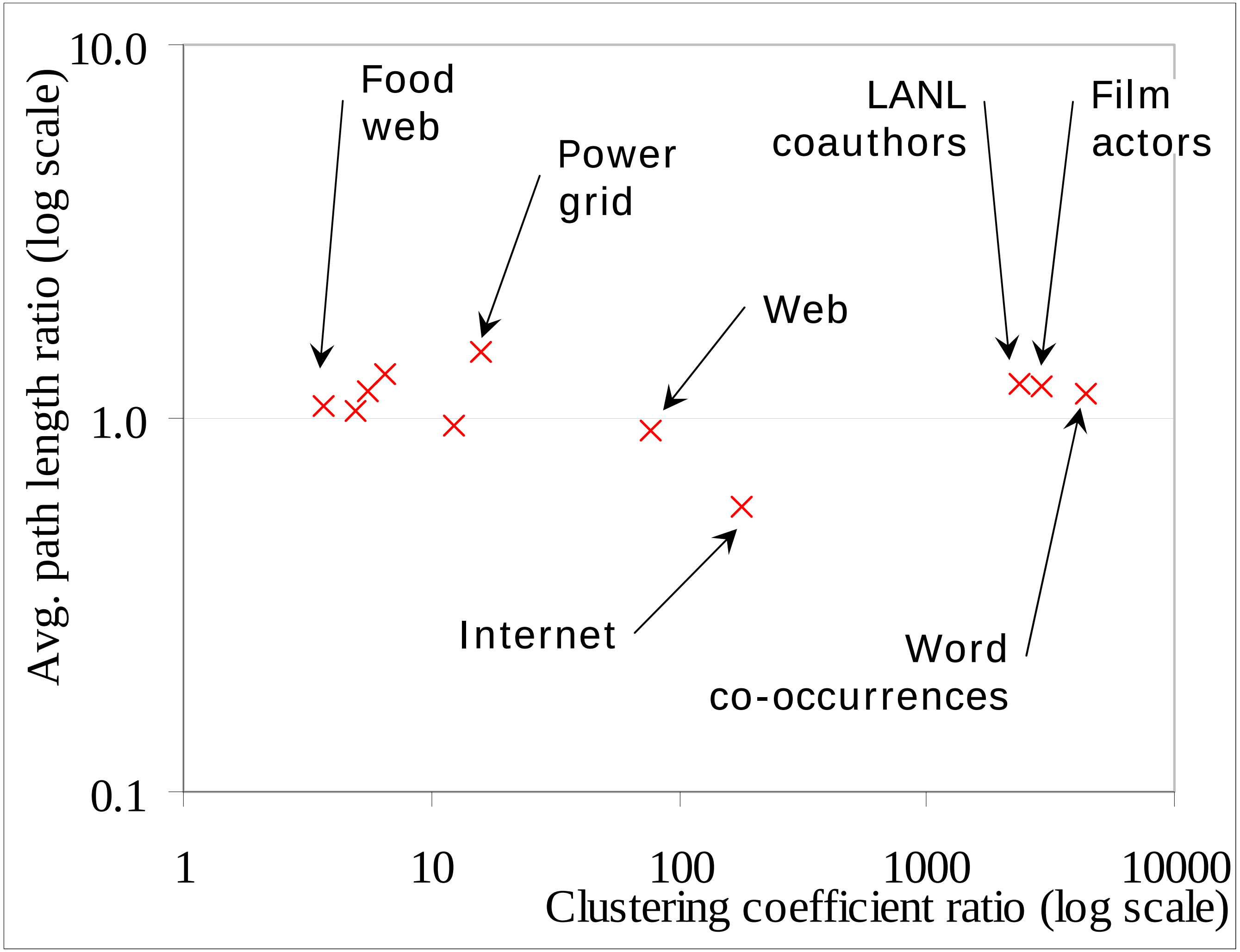
Large clustering coefficient

Small-World Graphs

- Small path length, large clustering coefficient
  - Typically compared against random graphs
- Think of:
  - “It’s a small world!”
  - “Six degrees of separation”
- Milgram’s experiments in the 60s
- Guare’s play “Six Degrees of Separation”



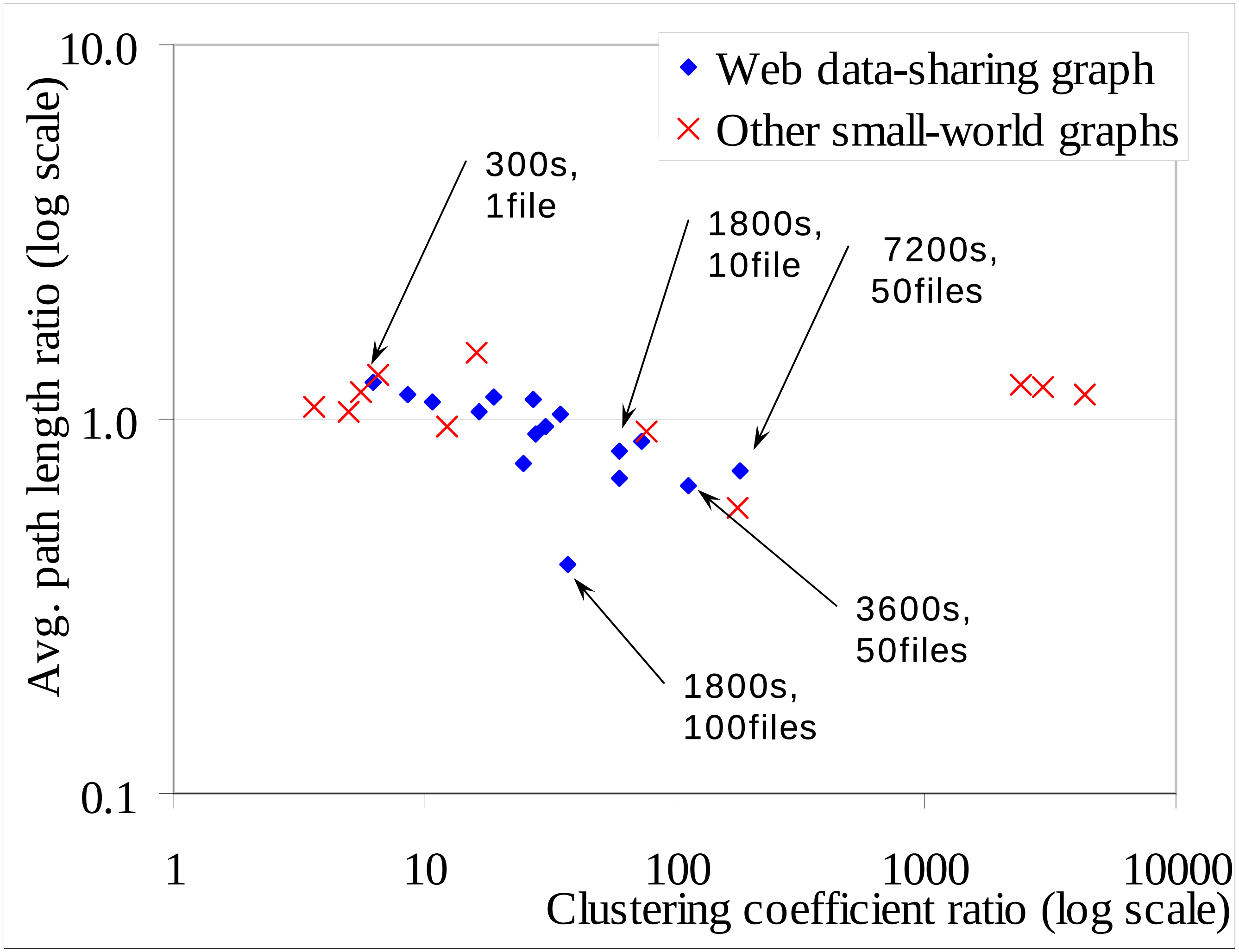
Other Small Worlds



D. J. Watts and S. H. Strogatz, *Collective dynamics of small-world networks*. Nature, 393:440-442, 1998

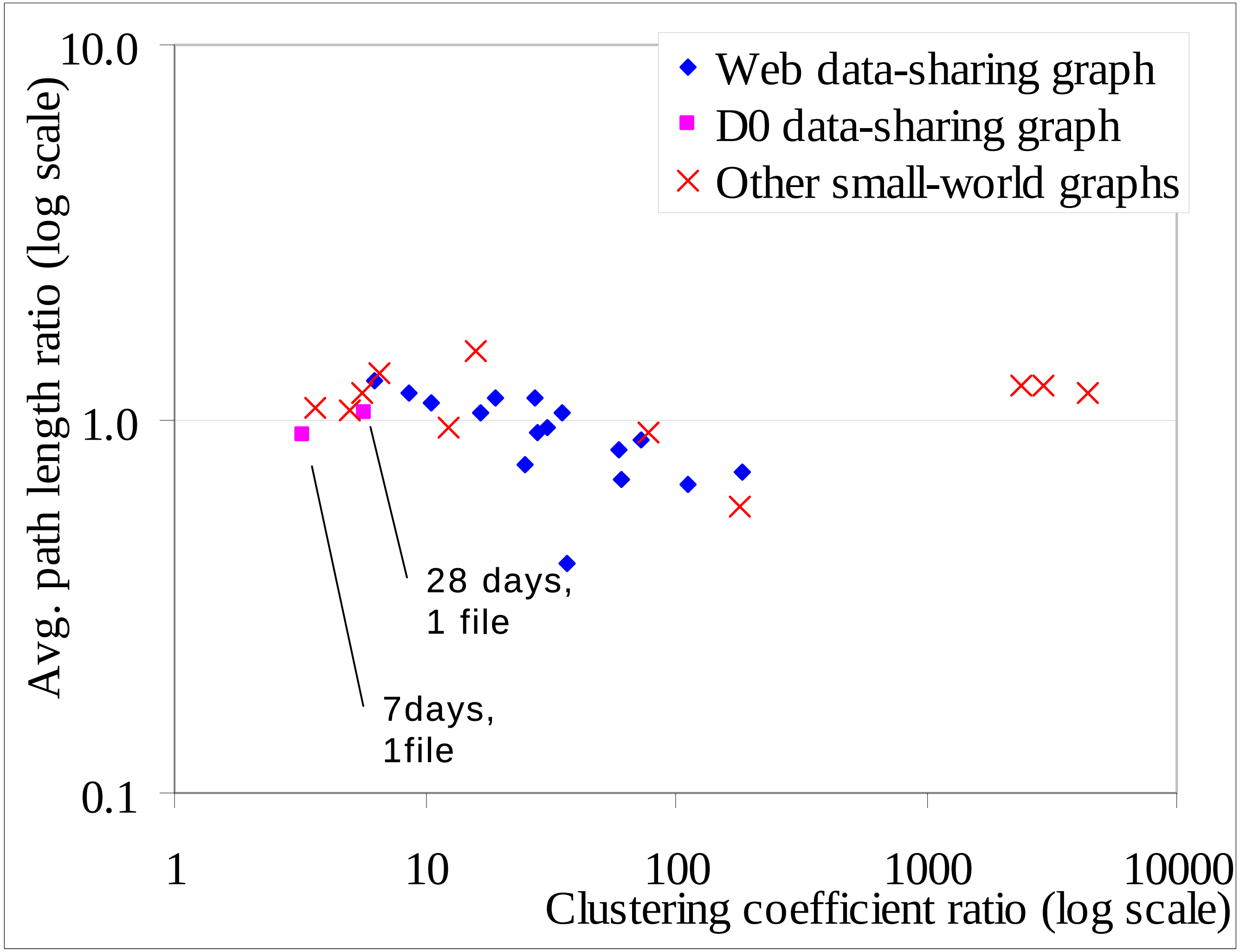
R. Albert and A.-L. Barabási, *Statistical mechanics of complex networks*, R. Modern Physics 74, 47 (2002).

Web Data-Sharing Graphs

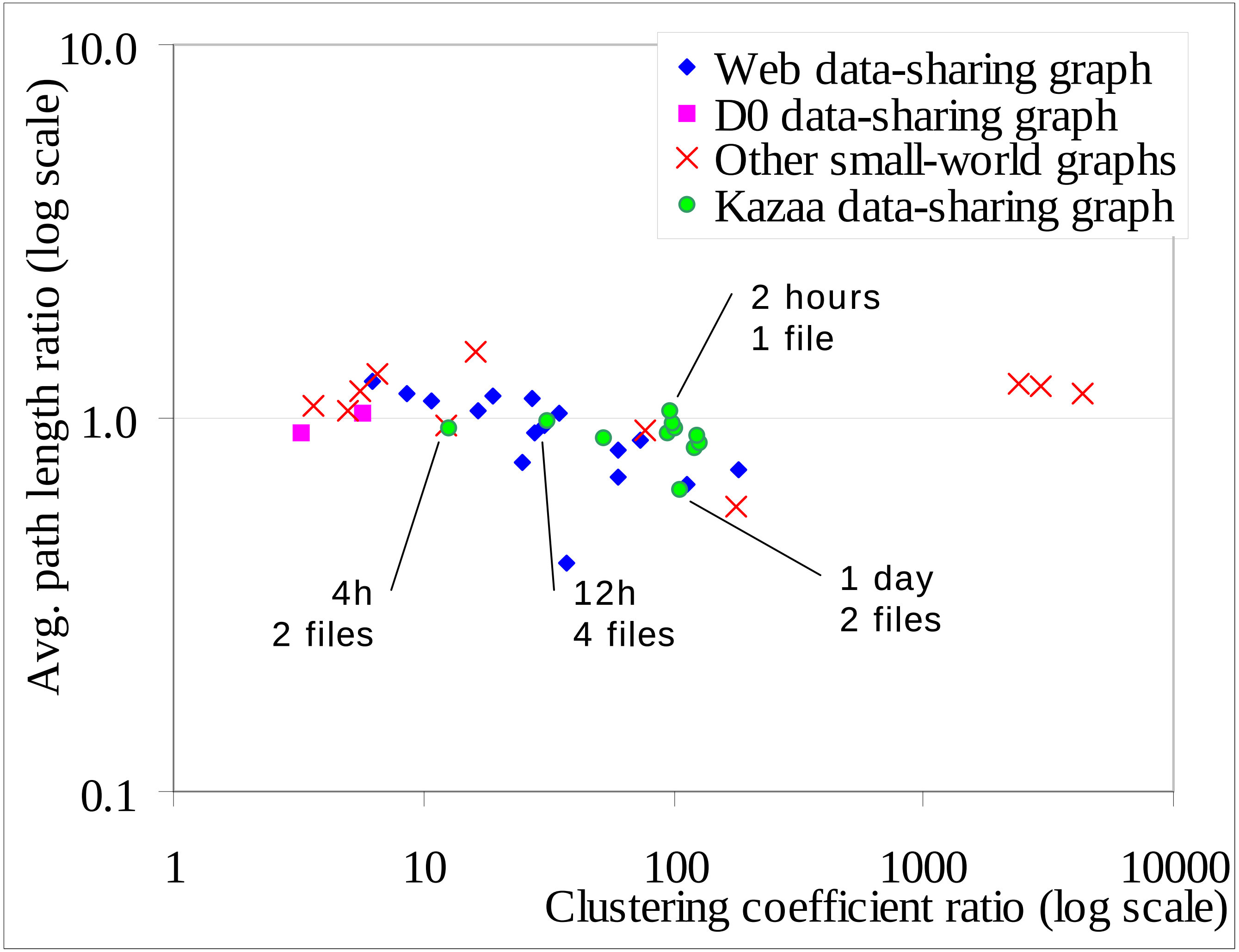


Data-Sharing Relationships in the Web, Iamnitchi, Ripeanu, and Foster, WWW'03

DØ Data-Sharing Graphs

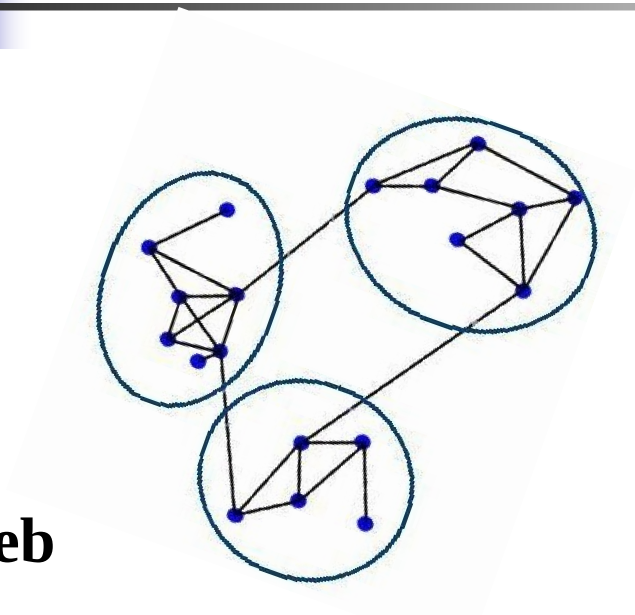


KaZaA Data-Sharing Graphs

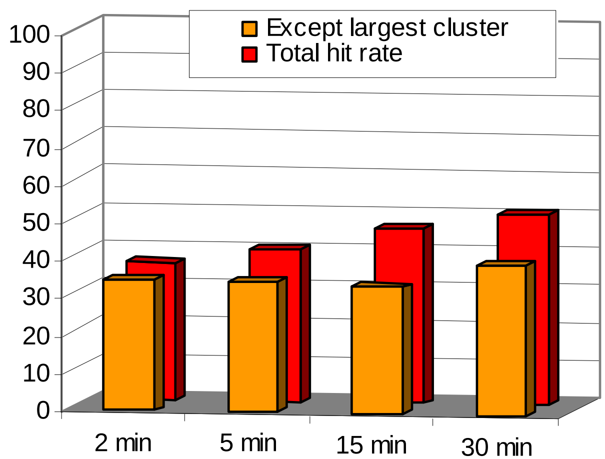


Small-World File-Sharing Communities, Iamnitchi, Ripeanu, and Foster, Infocom '04

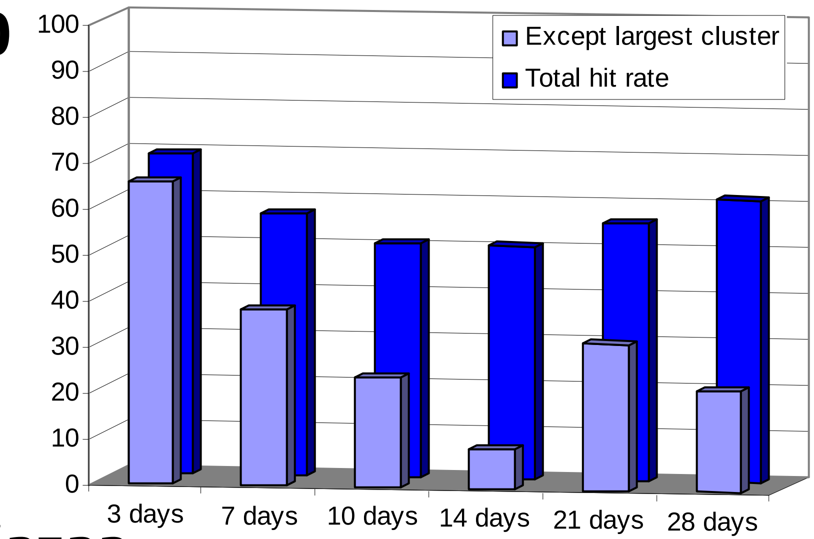
Interest-Aware Data Dissemination



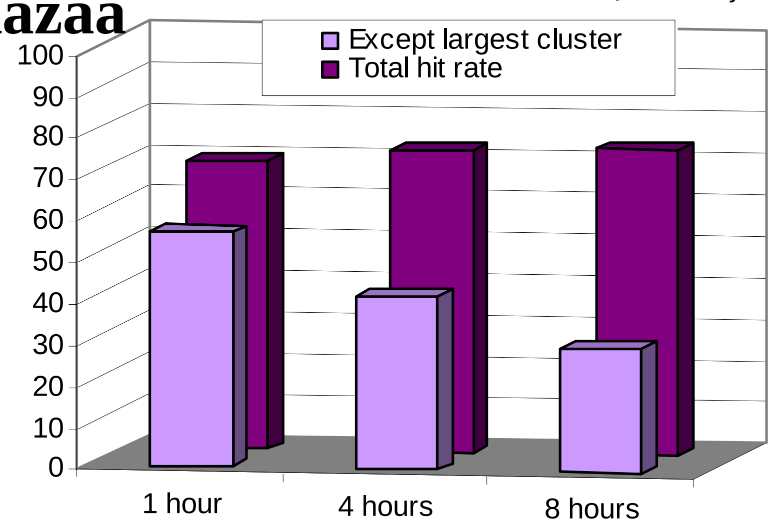
Web



D0



Kazaa



Interest-Aware Information Dissemination in Small-World Communities, Iamnitchi and Foster, HPDC'05

Amazon's Simple Storage Service: Cost Evaluation for D0

Work with Mayur Palankar,
Ayodele Onibokun (USF) and
Matei Ripeanu (UBC)

Amazon's Simple Storage Service

- Novel storage 'utility':
  - Direct access to storage
- Self-defined performance targets:
  - Scalable, infinite data durability, 99.99% availability, fast data access
- Pay-as-you go pricing:
  - $0.15/month/GB stored and $0.20/GB transferred
  - Recently updated pricing scheme

Is offloading data storage from an in-house mass-storage system to S3 feasible and cost-effective for scientists?

Amazon S3 Architecture

- Two level namespace
  - Buckets (think directories)
    - Unique names
    - Two goals: data organization and charging
  - Data objects
    - Opaque object (max 5GB)
    - Metadata (attribute-value, up to 4K)

- Functionality
  - Simple put/get functionality
  - Limited search functionality
  - Objects are immutable, cannot be renamed

- Data access protocols
  - SOAP
  - REST
  - BitTorrent

S3 Architecture (...cont)

- Security
  - Identities
    - Assigned by S3 when initial contract is ‘signed’
  - Authentication
    - Public/private key scheme
    - But private key is generated by Amazon!
  - Access control
    - Access control lists (limited to 100 principals)
    - ACL attributes
      - FullControl,
      - Read & Write (for buckets only for writes)
      - ReadACL & WriteACL (for buckets or objects)
  - Auditing (pseudo)
    - S3 can provide a log record

Approach

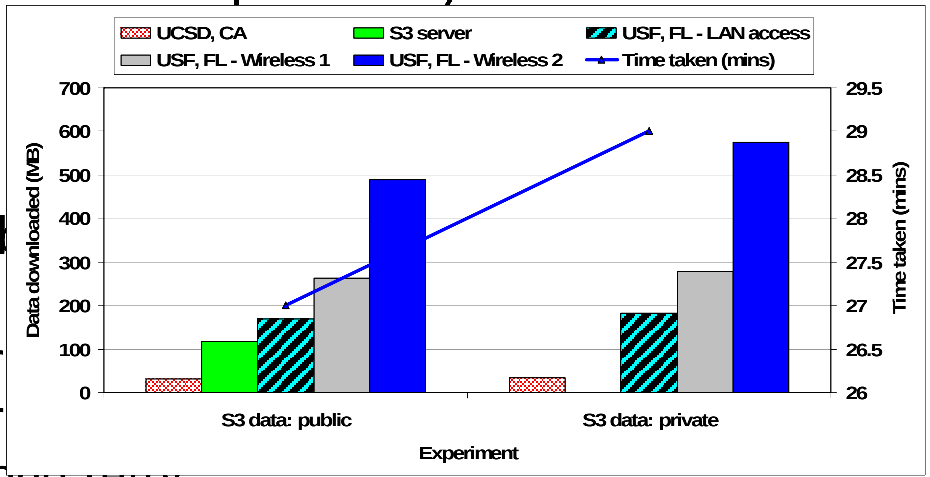
- Characterize S3
  - Does it live up to its own expectations?
- Estimate the performance and cost of a representative scientific application (DZero) in this context
- Is the functionality provided adequate?

S3 characterization methodology

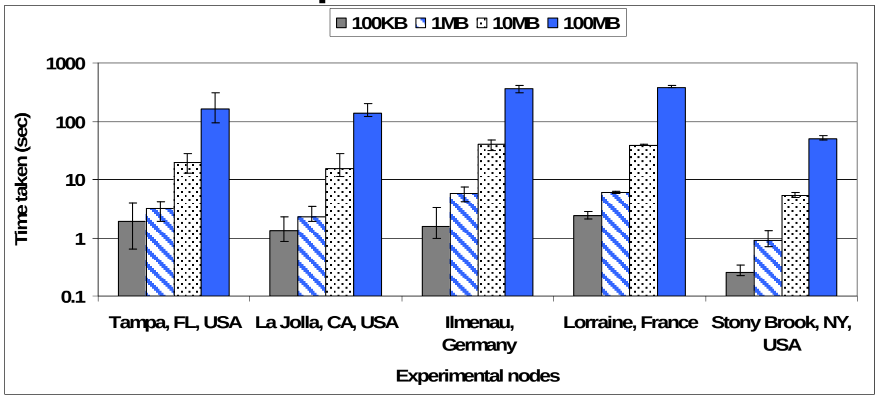
- Black-box approach using PlanetLab nodes to estimate:
  - durability,
  - availability,
  - access performance,
  - the effect of BitTorrent on cost savings
- Isolate local failures

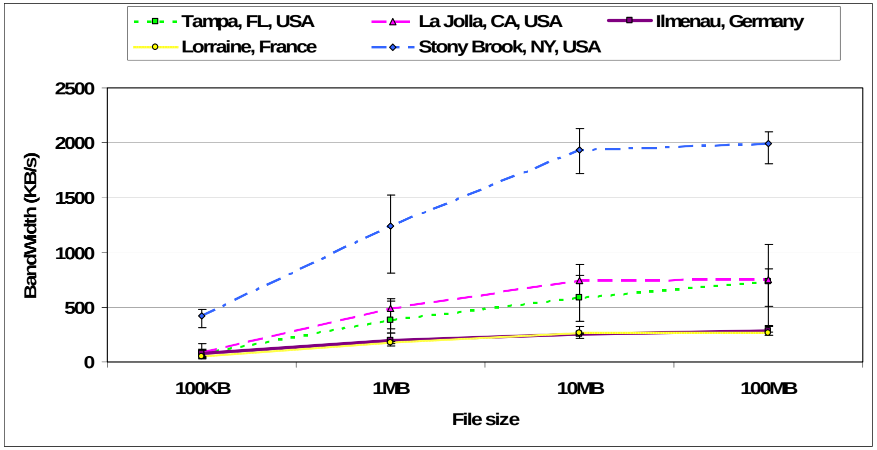
S3 Evaluation

- Durability
  - Perfect (but based on limited scale experiment)
- Availability
  - Four weeks of traces, about
    PlanetLab nodes
  - Retry protocol, exponential
  - ‘Cleaned’ data
    - 99.03% availability after or
    - 99.55% availability after fir
    - 100% availability after second retry



- Access performance





S3 Evaluation: Security

- Risks
  - Traditional risks with distributed storage are still a concern:
    - Permanent data loss,
    - Temporary data unavailability (DoS),
    - Loss of confidentiality
    - Malicious or erroneous data modifications
  - New risk: direct monetary loss
    - Magnified as there is no built-in solution to limit loss
- Security scheme's big advantage: it's simple
- … but has limitations
  - Access control
    - Hard to use ACLs in large systems – needs at least groups
    - ACLs limited to 100 principals
  - No support for fine grained delegation
  - Implicit trust between users and the service S3
    - No 'receipts'
    - No support for un-repudiabiliy
  - No tools to limit risk

S3 Evaluation: Cost

- Hypothetical scenario:
  - S3 used by a scientific community: The DZero Experiment
    - 375 TB data, 5.2 PB processed
- Costs
  - Scenario 1: All data stored at S3 and processed by DZero
    - Storage $675,000/year for storage ($.15/GB)
    - Transfer $462,222/year for transfer ($.20/GB. Now $.13-$.18/GB)
      - → $94,768 per month !
  - Scenario 2: Reducing transfer costs
    - Caching: With a 50TB cooperative cache → $66,329 per year in transfer costs
    - Using EC2 → No transfer costs but about 45K in compute costs.
  - Scenario 3: Reducing storage costs
    - Useful characteristic: data gets 'cold'
      - Throw away derived data
      - Archive old data – better with S3 support

Summary

- Workload characterization based on a HEP grid
  - Quantify scale (data processed, number of files)
  - Contradict traditional models
- Patterns can guide resource management
  - Filecules: caching, data replication
  - Small world data sharing: adaptive information dissemination, replica placement

Thank you.

Questions

- Storage costs for D0: how do they compared with S3 costs?
- Would you use a storage utility?
- What would you request from a storage utility provider:
  - Usage records: need to be private?
  - Benefits
- Other traces?

Other Performance Metrics